



APRENDERAPROGRAMAR.COM

QUERYSELECTORALL
JAVASCRIPT Y
QUERYSELECTOR.
ACCEDER A ELEMENTOS
POR SELECTORES CSS.
EJEMPLOS (CU01134E)

Sección: Cursos

Categoría: Tutorial básico del programador web: JavaScript desde cero

Fecha revisión: 2029

Resumen: Entrega nº34 del Tutorial básico “JavaScript desde cero”.

Autor: César Krall

QUERYSELECTORALL Y QUERYSELECTOR JAVASCRIPT

Hemos visto que JavaScript provee métodos para acceder y manipular nodos del DOM, en concreto `getElementById`, `getElementsByTagName` y `getElementsByName`, que nos permiten acceder a un elemento por su valor de atributo `id`, o a la colección de elementos cuya etiqueta es de un determinado tipo o tiene como atributo `name` un valor concreto. Veremos ahora métodos que nos permiten acceder a cualquier nodo identificado por un selector CSS específico.



QUERYSELECTORALL Y QUERYSELECTOR

Si recordamos la sintaxis CSS, a través de selectores podíamos definir de forma muy específica cómo se debían aplicar estilos dentro de nuestra página web. Por ejemplo a través de la regla:

```
#menu1 div.destacado {background-color: yellow;}
```

Se consigue que los elementos `a` (links) dentro de elementos `div` cuyo atributo `class` es "destacado" y están dentro de un elemento con `id` `menu1` se muestren con color de fondo amarillo.

El selector `#menu1 div.destacado a` es un selector complejo que no resultaría alcanzable con las técnicas de acceso a nodos que hemos visto hasta el momento. Por ello surgen los métodos `querySelector` y `querySelectorAll`, que permiten acceder a nodos a través de un selector CSS.

La sintaxis a emplear es la siguiente:

```
document.querySelectorAll("aquí el selector CSS");
```

Por ejemplo `document.querySelectorAll("#calculadora .teclas span.operador");`

La sintaxis a incluir dentro de las comillas para especificar el selector es exactamente la misma sintaxis que empleamos con CSS. Dentro de la especificación se pueden indicar varios selectores separados por comas, al igual que hacemos con CSS para aplicar una misma regla a distintos elementos.

Esta instrucción nos devuelve un array de nodos del DOM que cumplen con la especificación del selector CSS. Los índices del array serán 0, 1, 2, ... hasta un número indeterminado (que dependerá del número de nodos que cumplan con el selector). Los nodos estarán ordenados según su orden de aparición en el árbol del DOM que define el documento HTML.

Frente al método general `querySelectorAll` existe otro método específico, `querySelector`, que devuelve un único nodo que cumple la especificación del selector. Este único nodo devuelto será el primero encontrado que cumpla con la especificación dentro del árbol del DOM.

La sintaxis es:

```
document.querySelector ("aquí el selector CSS ");
```

Típicamente el array de nodos (o el nodo individual) obtenido con estos métodos los almacenaremos dentro de una variable JavaScript para después realizar algún tipo de tratamiento. Por ejemplo:

```
var listaNodosOperadores = document.querySelectorAll("#calculadora .teclas span.operador");
```

```
var nodoligual = document.querySelector (".igual");
```

Una vez tenemos una colección de nodos en una variable, podemos recorrerla usando un bucle for. Por ejemplo:

```
for (var i=0; i<listaNodosOperadores.length; i++) {  
  
    // Ejecutar acciones sobre el nodo listaNodosOperadores[i].  
  
}
```

Los métodos querySelector se pueden aplicar sobre document como hemos visto (para seleccionar todos los nodos hijos del nodo raíz que cumplen un criterio), pero también sobre un nodo especificado obtenido de otra manera. Por ejemplo:

```
var listaNodos = document.getElementById("calculadora").querySelectorAll(".teclas span.numero");
```

Aquí primero se obtiene el nodo cuyo id es "calculadora" y después se obtienen todos los nodos hijos de calculadora cuyo que son elementos span con class="numero" que se encuentran dentro de un elemento con un valor class="teclas"

En este caso lo indicado anteriormente se puede escribir también como:

```
var listaNodosOperadores = document.querySelectorAll("#calculadora .teclas span.operador");
```

Esta última forma de escritura resulta quizás más clara, y por eso la consideraremos preferible cuando podamos elegir entre usar una forma u otra.

Escribe y prueba el siguiente código, donde vemos un ejemplo de uso de querySelector y de querySelectorAll. Ten en cuenta que en algunos navegadores, en especial los más antiguos, quizás no se obtengan los resultados deseados.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<style type="text/css">
body {font-family: sans-serif; text-align:center;}
#calculadora { font: bold 14px Arial,sans-serif; background-color: #9DD2EA;
border-radius: 3px; height: auto; margin: 0 auto; padding: 20px 20px 9px; width: 295px;
}
.parteSuperior .pantalla {
background-color: rgba(0, 0, 0, 0.2); border-radius: 3px; color: #FFFFFF; float: right; font-size: 17px;
height: 40px; letter-spacing: 1px; line-height: 40px; padding: 0 10px; text-align: right; width: 180px;
}
.teclas, .parteSuperior {overflow: hidden; }
.teclas span, .parteSuperior span.limpiar { background-color: #FFFFFF; border-radius: 3px; color: #888888;
cursor: pointer; float: left; height: 36px; line-height: 36px; margin: 0 7px 11px 0;
text-align: center; transition: all 0.4s ease 0s; width: 66px;
}
.parteSuperior span.limpiar { background-color:#FF7C87;}
.teclas span.operador { background-color: #FFDAB9; margin-right: 0; }
.teclas span.igual { background-color: #F1FF92; color: #888E5F;}
.parteSuperior span.limpiar, { background-color: #FF9FA8; color: #FFFFFF; }
.teclas span:hover, .teclas span.igual:hover, .parteSuperior span.limpiar:hover { background-color: #9C89F6;
color: #FFFFFF; }

#contenedor {width:285px;margin: 40px auto; overflow:hidden;}
#marcaNumeros, #marcaOperadores {padding:15px; width: 90px; display: inline-block;
margin: 10px; cursor: pointer; color: white; border-radius: 40px; background: rgb(202, 60, 60);}
#marcaNumeros:hover, #marcaOperadores:hover {background: rgb(66, 184, 221);}
</style>

<script type="text/javascript">

function marcarOperadores() {
var listaNodosOperadores = document.querySelectorAll("#calculadora .teclas span.operador");
var nodoligual = document.querySelector (".igual");
for (var i=0; i<listaNodosOperadores.length; i++) {
listaNodosOperadores[i].style.backgroundColor = "yellow";
listaNodosOperadores[i].style.color = "black";
listaNodosOperadores[i].style.border = "solid 1px";
}
nodoligual.style.backgroundColor = "yellow";
nodoligual.style.color = "black";
nodoligual.style.border = "solid 1px";
}

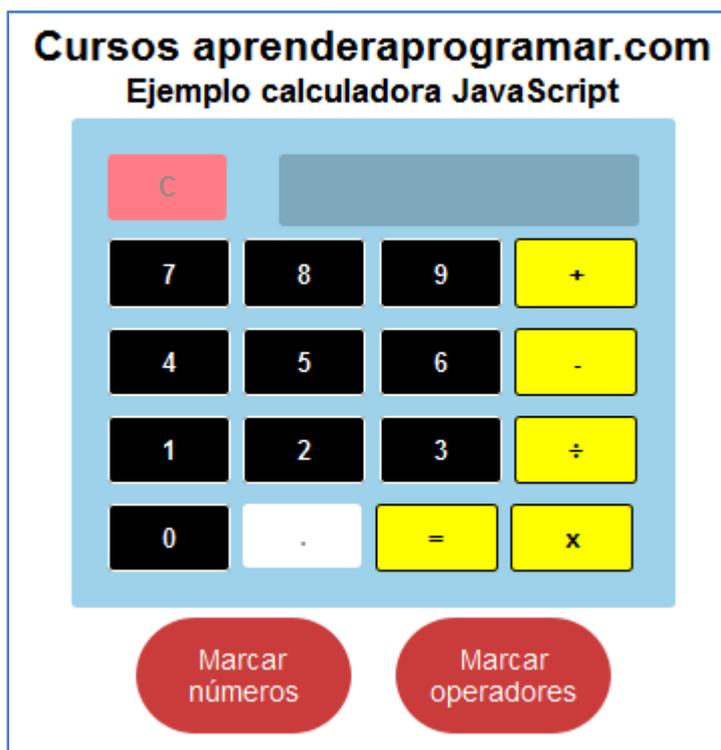
function marcarNumeros() {
var listaNodosNumeros = document.querySelectorAll("#calculadora .teclas span.numero");
for (var i=0; i<listaNodosNumeros.length; i++) {
listaNodosNumeros[i].style.backgroundColor = "black";
listaNodosNumeros[i].style.color = "white";
listaNodosNumeros[i].style.border = "solid 1px";
}
}
</script>
</head>
```

```
<body>
<div id="cabecera">
<h2>Cursos aprenderaprogramar.com</h2>
<h3>Ejemplo calculadora JavaScript</h3>
</div>
<!-- Calculadora -->
<div id="calculadora">
  <!-- pantalla y tecla limpiar -->
  <div class="parteSuperior">
    <span class="limpiar" onclick="pulsada('C')>C</span>
    <div class="pantalla"> </div> <!--Ojo tenemos un espacio para que exista nodeValue-->
  </div>

  <div class="teclas">
    <!-- operadores y otras teclas -->
    <span class = "numero" onclick="pulsada('7')>7</span>
    <span class = "numero" onclick="pulsada('8')>8</span>
    <span class = "numero" onclick="pulsada('9')>9</span>
    <span class="operador" onclick="pulsada('+')>+</span>
    <span class = "numero" onclick="pulsada('4')>4</span>
    <span class = "numero" onclick="pulsada('5')>5</span>
    <span class = "numero" onclick="pulsada('6')>6</span>
    <span class="operador" onclick="pulsada('-')>-</span>
    <span class = "numero" onclick="pulsada('1')>1</span>
    <span class = "numero" onclick="pulsada('2')>2</span>
    <span class = "numero" onclick="pulsada('3')>3</span>
    <span class="operador" onclick="pulsada('/')>÷</span>
    <span class = "numero" onclick="pulsada('0')>0</span>
    <span class ="punto" onclick="pulsada('.')>.</span>
    <span class="igual" onclick="pulsada('=')>=</span>
    <span class="operador" onclick="pulsada('*')>x</span>
  </div>
</div>
<div id="contenedor">
<div id ="marcaNumeros" onclick="marcarNumeros()"> Marcar números </div>
<div id="marcaOperadores" onclick="marcarOperadores()" >Marcar operadores</div>
</div>
</body>
</html>
```

El resultado esperado es que se muestre una calculadora y en su parte inferior dos botones: “Marcar números” y “Marcar operadores”. Al pulsar sobre “Marcar números” las teclas correspondientes a los números de la calculadora deberán verse de color negro con los números de color blanco. Al pulsar sobre “Marcar operadores” las teclas correspondientes a los operadores (+, -, =, etc.) deberán verse de color amarillo con los operadores de color negro.

Fíjate en el código: la selección de las teclas afectadas y el cambio de estilos se hace gracias al uso de los operadores querySelectorAll y querySelector de JavaScript.



EJERCICIO

1) Modifica el código del ejemplo de la calculadora JavaScript para que además de tener dos botones que permitan marcar números y operadores con distintos colores tenga:

- Un botón que permita marcar la tecla C con color rojo fuerte y símbolo C en blanco.
- Un botón que permita marcar la tecla con el punto con color violeta fuerte y símbolo . en blanco.
- Un botón que permita marcar la pantalla visor de la calculadora en azul oscuro.
- Un botón que permita limpiar todo lo que se haya marcado y volver a la situación original (es decir, que la calculadora se vea tal y como se veía inicialmente).

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01135E

Acceso al curso completo en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:
http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=78&Itemid=206